Problem Statement-7

```java
package arrays;
public class Doublell {
      Node head;
class Node
{
                    int data;
                    Node prev;
                    Node next;
Node(int d)
{
data = d;
}
      }
public void push(int new_data)
      {
Node new_Node = new Node(new_data);
new_Node.next = head;
                    new_Node.prev = null;
if (head != null)
                          head.prev = new_Node;
head = new_Node;
      }
public void InsertAfter(Node prev_Node, int new_data)
      {
if (prev_Node == null)
{
                          System.out.println("The given previous node cannot be
NULL ");
                          return;
                }
Node new_node = new Node(new_data);
new_node.next = prev_Node.next;
prev_Node.next = new_node;
new_node.prev = prev_Node;
if (new_node.next != null)
                          new_node.next.prev = new_node;
      }
      void append(int new_data)
      {
Node new_node = new Node(new_data);
          Node last = head;
new_node.next = null;
if (head == null)
{
                          new_node.prev = null;
                          head = new_node;
                          return;
```
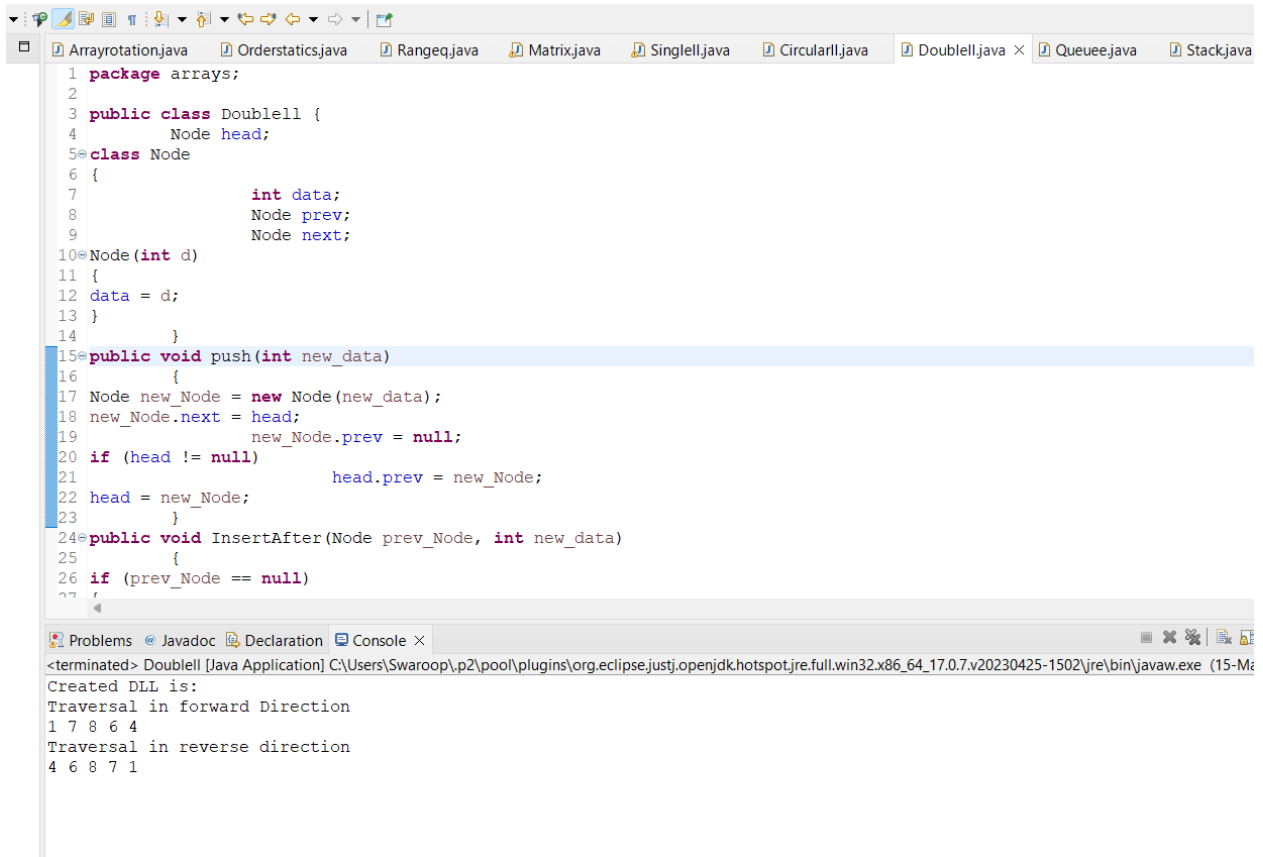
```java
                }
while (last.next != null)
                        last = last.next;
last.next = new_node;
new_node.prev = last;
        }
public void printlist(Node node)
        {
                Node last = null;
                System.out.println("Traversal in forward Direction");
                while (node != null)
{
                        System.out.print(node.data + " ");
                        last = node;
                        node = node.next;
                }
                System.out.println();
                System.out.println("Traversal in reverse direction");
                while (last != null)
{
                        System.out.print(last.data + " ");
                        last = last.prev;
                }
        }
public static void main(String[] args)
        {
Doublell dll = new Doublell();
dll.append(6);
dll.push(7);
dll.push(1);
dll.append(4);
dll.InsertAfter(dll.head.next, 8);
            System.out.println("Created DLL is: ");
                dll.printlist(dll.head);
        }
}
```

```java
1 package arrays;
2
3 public class Doublell {
4         Node head;
5 class Node
6 {
7             int data;
8             Node prev;
9             Node next;
10 Node(int d)
11 {
12 data = d;
13 }
14         }
15 public void push(int new_data)
16       {
17 Node new_Node = new Node(new_data);
18 new_Node.next = head;
19           new_Node.prev = null;
20 if (head != null)
21               head.prev = new_Node;
22 head = new_Node;
23       }
24 public void InsertAfter(Node prev_Node, int new_data)
25       {
26 if (prev_Node == null)
27 {
```

```
<terminated> Doublell [Java Application] C:\Users\Swaroop\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.7.v20230425-1502\jre\bin\javaw.exe  (15-M
Created DLL is:
Traversal in forward Direction
1 7 8 6 4
Traversal in reverse direction
4 6 8 7 1
```